

■ Lista de 200 Exercícios de Java para Iniciantes

Módulo 1 – Fundamentos (20 exercícios)

1. Escreva um programa que mostre 'Olá, Mundo!' no console.
2. Crie um programa que peça seu nome e exiba uma saudação.
3. Leia dois números e exiba a soma.
4. Leia dois números e exiba a subtração.
5. Leia dois números e exiba a multiplicação.
6. Leia dois números e exiba a divisão.
7. Calcule a média de três números.
8. Leia um número e mostre o dobro dele.
9. Leia um número e mostre o quadrado dele.
10. Converta metros em centímetros.
11. Converta Celsius em Fahrenheit.
12. Converta Fahrenheit em Celsius.
13. Calcule a área de um quadrado.
14. Calcule a área de um triângulo.
15. Calcule a área de um círculo.
16. Calcule o perímetro de um retângulo.
17. Leia um número e exiba o sucessor e antecessor.
18. Leia a idade e mostre em meses.
19. Leia o valor de um produto e mostre com 10% de desconto.
20. Leia um salário e mostre com aumento de 15%.

Módulo 2 – Estruturas Condicionais (30 exercícios)

1. Verifique se um número é positivo ou negativo.
2. Verifique se um número é par ou ímpar.
3. Leia duas notas e informe se o aluno foi aprovado (média ≥ 7).
4. Verifique se um ano é bissexto.
5. Calcule o IMC e classifique (baixo peso, normal, sobrepeso, obeso).
6. Leia a idade e diga se pode votar.
7. Leia a idade e diga se pode dirigir.
8. Compare dois números e diga qual é maior.
9. Verifique se três lados podem formar um triângulo.
10. Classifique um triângulo em equilátero, isósceles ou escaleno.
11. Verifique se uma letra é vogal ou consoante.
12. Verifique se uma senha digitada é igual à '1234'.
13. Calcule o preço do ingresso (meia ou inteira).
14. Leia um número de 1 a 7 e mostre o dia da semana correspondente.
15. Verifique se um número é múltiplo de 3 e 5 ao mesmo tempo.
16. Verifique se um número está no intervalo entre 10 e 50.
17. Leia um caractere e verifique se é letra, número ou símbolo.
18. Calcule a média de três notas e dê conceito (A, B, C, D).
19. Leia dois números e um operador (+, -, *, /) e calcule.
20. Verifique se uma pessoa tem direito à aposentadoria (idade ≥ 65 ou tempo de contribuição ≥ 30 anos).
21. Leia um número de 1 a 12 e mostre o mês correspondente.
22. Calcule a tarifa de energia com base no consumo.
23. Calcule a multa por excesso de velocidade.
24. Leia um número e verifique se é primo (simplificado).
25. Determine o maior de três números.
26. Determine o menor de três números.
27. Calcule o valor da passagem com base na distância (até 200 km preço fixo, acima disso preço/km).
28. Leia dois horários e diga qual vem primeiro.
29. Verifique se um número é perfeito (soma dos divisores = número).
30. Leia um número e verifique se é palíndromo.

Módulo 3 – Laços de Repetição (30 exercícios)

1. Mostre os números de 1 a 10.
2. Mostre os números pares de 1 a 50.
3. Mostre os números ímpares de 1 a 50.
4. Calcule a soma dos números de 1 a 100.
5. Calcule a soma dos números pares de 1 a 100.
6. Calcule a soma dos números ímpares de 1 a 100.
7. Leia um número N e mostre os números de 1 até N.
8. Leia um número N e mostre a tabuada dele.
9. Leia 10 números e mostre a soma.
10. Leia 10 números e mostre a média.
11. Leia 10 números e mostre o maior.
12. Leia 10 números e mostre o menor.
13. Leia números até que o usuário digite 0 e some-os.
14. Leia números até que o usuário digite negativo e calcule a média.
15. Leia um número N e calcule o fatorial.
16. Mostre a sequência de Fibonacci até N termos.
17. Calcule a soma dos dígitos de um número.
18. Mostre os múltiplos de 5 até 100.
19. Mostre os divisores de um número.
20. Conte quantos números pares foram digitados.
21. Conte quantos números ímpares foram digitados.
22. Leia 20 números e diga quantos são positivos.
23. Leia 20 números e diga quantos são negativos.
24. Calcule a média de notas de uma turma até digitar -1.
25. Mostre um triângulo de asteriscos (for).
26. Mostre um quadrado de asteriscos.
27. Mostre a tabuada de 1 a 10.
28. Leia um número N e mostre os N primeiros números primos.
29. Calcule o MDC de dois números.
30. Calcule o MMC de dois números.

Módulo 4 – Vetores (20 exercícios)

1. Leia um vetor de 10 posições e mostre os valores.
2. Leia um vetor e mostre os valores na ordem inversa.
3. Leia um vetor de 10 números e mostre a soma.
4. Leia um vetor de 10 números e mostre a média.
5. Mostre o maior elemento de um vetor.
6. Mostre o menor elemento de um vetor.
7. Conte quantos pares há em um vetor.
8. Conte quantos ímpares há em um vetor.
9. Leia um vetor e substitua números negativos por 0.
10. Leia dois vetores e some-os.
11. Leia dois vetores e multiplique-os (produto escalar).
12. Leia um vetor e mostre somente os números pares.
13. Leia um vetor e mostre somente os números ímpares.
14. Ordene um vetor em ordem crescente.
15. Ordene um vetor em ordem decrescente.
16. Leia um vetor e verifique se há números repetidos.
17. Leia um vetor e encontre a moda (valor que mais se repete).
18. Leia um vetor e encontre a mediana.
19. Gere 10 números aleatórios e armazene em um vetor.
20. Crie um programa que embaralhe os elementos de um vetor.

Módulo 5 – Matrizes (20 exercícios)

1. Leia uma matriz 3x3 e mostre os valores.
2. Leia uma matriz e mostre a soma dos elementos.
3. Leia uma matriz e mostre a média dos elementos.
4. Leia uma matriz e mostre a diagonal principal.
5. Leia uma matriz e mostre a diagonal secundária.
6. Leia uma matriz e calcule a soma de cada linha.
7. Leia uma matriz e calcule a soma de cada coluna.
8. Leia uma matriz e encontre o maior valor.
9. Leia uma matriz e encontre o menor valor.
10. Leia duas matrizes e some-as.
11. Leia duas matrizes e multiplique-as.
12. Leia uma matriz e mostre os elementos pares.
13. Leia uma matriz e mostre os elementos ímpares.
14. Troque duas linhas de uma matriz.
15. Troque duas colunas de uma matriz.
16. Leia uma matriz e verifique se é simétrica.
17. Leia uma matriz identidade e mostre.
18. Gere uma matriz identidade NxN.
19. Gere uma matriz transposta.
20. Gere uma matriz com números aleatórios.

Módulo 6 – Strings (20 exercícios)

1. Leia uma string e mostre em maiúsculas.
2. Leia uma string e mostre em minúsculas.
3. Leia uma string e mostre a quantidade de caracteres.
4. Leia uma string e mostre a quantidade de vogais.
5. Leia uma string e mostre a quantidade de consoantes.
6. Leia uma string e verifique se é palíndromo.
7. Leia uma string e conte quantos espaços existem.
8. Leia uma string e substitua espaços por ' _ '.
9. Leia uma string e conte quantas vezes aparece a letra 'a'.
10. Leia uma string e inverta.
11. Leia uma string e verifique se contém apenas letras.
12. Leia uma string e verifique se contém apenas números.
13. Leia uma string e conte quantas palavras tem.
14. Leia uma frase e mostre a primeira palavra.
15. Leia uma frase e mostre a última palavra.
16. Leia duas strings e compare se são iguais.
17. Leia duas strings e concatene.
18. Leia uma string e mostre sem as vogais.
19. Leia uma string e mostre sem as consoantes.
20. Leia uma string e embaralhe os caracteres.

Módulo 7 – Programação Orientada a Objetos (40 exercícios)

1. Crie uma classe Pessoa com atributos nome e idade.
2. Crie uma classe Aluno que herda de Pessoa.
3. Crie uma classe Funcionario com salário e cargo.
4. Crie uma classe Produto com nome e preço.
5. Crie uma classe ContaBancaria com saldo.
6. Crie métodos para depositar e sacar na ContaBancaria.
7. Crie uma classe Carro com atributos modelo e ano.
8. Adicione método para acelerar o carro.
9. Adicione método para frear o carro.
10. Crie uma classe Retângulo com métodos para calcular área e perímetro.
11. Crie uma classe Círculo com método para calcular área.
12. Crie uma classe Triângulo e calcule a área.
13. Crie uma classe Aluno com lista de notas.
14. Calcule a média das notas do aluno.
15. Verifique se o aluno foi aprovado.
16. Crie uma classe Livro com título e autor.
17. Crie uma classe Biblioteca que armazena livros.
18. Crie uma classe Agenda que armazena contatos.
19. Crie uma classe Contato com nome e telefone.
20. Crie uma classe Calculadora com operações básicas.
21. Crie uma classe Animal com método emitirSom.
22. Crie subclasses Cachorro e Gato que sobrescrevem emitirSom.
23. Crie uma interface Imprimivel com método imprimir().
24. Faça uma classe Relatorio que implementa Imprimivel.
25. Crie uma classe abstrata Figura com método area().
26. Crie subclasses Quadrado e Circulo que implementam area().
27. Crie uma classe Banco que armazena contas bancárias.
28. Implemente transferência entre contas.
29. Crie uma classe Jogo com nome e gênero.
30. Crie uma classe Loja que vende produtos.
31. Crie uma classe Pedido com lista de produtos.
32. Calcule o valor total de um pedido.
33. Crie uma classe Aluno com matrícula e curso.
34. Crie uma classe Escola que armazena alunos.
35. Crie uma classe Turma que possui lista de alunos.
36. Crie uma classe Professor com disciplina.

37. Associe Professor e Turma.
38. Crie uma classe Empresa com lista de funcionários.
39. Crie uma classe Projeto com lista de funcionários.
40. Associe Funcionário a Projeto.

Módulo 8 – Exercícios Extras e Desafios (20 exercícios)

1. Crie um jogo de adivinhação (1 a 100).
2. Crie um jogo de pedra, papel e tesoura.
3. Crie um jogo da forca.
4. Crie um jogo de dados.
5. Crie um cronômetro em segundos.
6. Crie uma calculadora com menu interativo.
7. Crie um conversor de moedas.
8. Crie um conversor de temperaturas.
9. Crie um simulador de caixa eletrônico.
10. Crie um sistema de cadastro de alunos.
11. Crie um sistema de cadastro de produtos.
12. Crie um sistema de cadastro de clientes.
13. Crie um sistema de reservas de hotel.
14. Crie um sistema de vendas.
15. Crie um sistema simples de login e senha.
16. Crie um sistema de votação.
17. Crie um sistema de controle de estoque.
18. Crie um sistema de notas escolares.
19. Crie um sistema de agenda pessoal.
20. Crie um sistema de biblioteca com empréstimo de livros.

Módulo 1 – Conceitos Básicos de Classes e Objetos (1 a 40)

1. Crie uma classe `Pessoa` com atributos `nome` e `idade` . Crie um objeto e exiba os valores.
2. Crie uma classe `Carro` com atributos `marca` e `modelo` . Inicialize e exiba os dados.
3. Crie uma classe `Livro` com atributos `titulo` e `autor` .
4. Crie um método `imprimirDetalhes()` na classe `Livro` .
5. Crie uma classe `Aluno` com atributos `nome` e `matricula` . Crie dois objetos e compare.
6. Crie uma classe `Produto` com atributos `nome` e `preco` . Adicione método para aplicar desconto.
7. Crie uma classe `Animal` com atributo `nome` e método `fazerSom()` .
8. Crie uma classe `ContaBancaria` com atributos `numero` e `saldo` .
9. Crie método `depositar(double valor)` na classe `ContaBancaria` .
10. Crie método `sacar(double valor)` na classe `ContaBancaria` .

11. Crie uma classe `Telefone` com atributos `marca` e `numero` .

12. Crie método `ligar()` na classe `Telefone` .

13. Crie método `receberLigacao()` na classe `Telefone` .

14. Crie uma classe `Endereco` com atributos `rua` , `numero` e `cidade` .

15. Crie método `imprimirEndereco()` na classe `Endereco` .

16. Crie uma classe `Funcionario` com atributos `nome` e `salario` .

17. Crie método `aumentarSalario(double percentual)` na classe `Funcionario` .

18. Crie uma classe `Pessoa` com construtores diferentes (sobrecarga).
19. Crie uma classe `Lampada` com atributo `ligada` e métodos `acender()` e `apagar()` .
20. Crie classe `Relogio` com método para exibir hora atual.
21. Crie classe `Aluno` e método `estudar()` .
22. Crie classe `Professor` com método `ensinar()` .
23. Crie classe `Carro` e método `acelerar()` .
24. Crie classe `Carro` e método `frear()` .
25. Crie classe `Casa` com atributo `cor` e método `pintar(String novaCor)` .
26. Crie classe `Computador` com método `ligar()` e `desligar()` .
27. Crie classe `Televisao` com método `aumentarVolume()` .

28. Crie método `diminuirVolume()` na classe `Televisao` .

29. Crie classe `Pedido` com atributos `codigo` e `valorTotal` .

30. Crie método `adicionarItem(String item, double preco)` na classe `Pedido` .

31. Crie classe `Banco` com método `adicionarConta(ContaBancaria c)` .

32. Crie método `listarContas()` na classe `Banco` .

33. Crie classe `Produto` com atributo `quantidade` .

34. Crie método `adicionarEstoque(int q)` na classe `Produto` .

35. Crie método `removerEstoque(int q)` na classe `Produto` .

36. Crie classe `Carro` com atributo `velocidadeAtual` .

37. Crie método `acelerar(int valor)` que aumenta a velocidade.

38. Crie método `frear(int valor)` que diminui a velocidade.

39. Crie classe `Pessoa` e método `aniversario()` que aumenta a idade.

40. Crie classe `Animal` e método `comer(String comida)` que imprime mensagem.

Módulo 2 – Encapsulamento e Getters/Setters (41 a 80)

1. Crie classe `Pessoa` com atributos privados `nome` e `idade` e crie `getters/setters`.
2. Crie classe `ContaBancaria` com `saldo` privado e métodos para acessar e modificar.
3. Crie classe `Produto` com atributos privados `nome` e `preco` .
4. Crie validação no setter do preço para não aceitar valores negativos.
5. Crie classe `Funcionario` com atributo `salario` privado.
6. Crie método para aumentar o salário, sem permitir valor negativo.
7. Crie classe `Aluno` com atributo `nota` .
8. Crie validação para nota entre 0 e 10.
9. Crie classe `Lampada` com atributo privado `ligada` .
10. Crie métodos `ligar()` e `desligar()` que modificam o atributo.

11. Crie classe `Carro` com velocidade privada e métodos de acesso.
12. Crie método `setVelocidade(int v)` com limite máximo de 200.
13. Crie classe `Endereco` com atributos privados e métodos de acesso.
14. Crie classe `Telefone` com atributo privado `numero`.
15. Crie método `setNumero(String n)` que verifica formato.
16. Crie classe `Produto` e método `imprimirInfo()` usando getters.
17. Crie classe `Banco` com lista de contas privadas e métodos de acesso.
18. Crie método `getSaldoTotal()` no Banco.
19. Crie classe `Pedido` com lista de itens privada.
20. Crie método para adicionar item usando encapsulamento.
21. Crie método para remover item do pedido usando encapsulamento.

22. Crie classe `Cliente` com CPF privado e validação no setter.

23. Crie classe `Funcionario` com atributo privado `departamento` .

24. Crie método `mudarDepartamento (String d)` com validação.

25. Crie classe `Carro` com atributo privado `combustivel` .

26. Crie método para abastecer e verificar limite máximo.

27. Crie classe `Pessoa` com atributo privado `altura` .

28. Crie método para alterar altura apenas se positiva.

29. Crie classe `Aluno` com matrícula privada.

30. Crie método para gerar matrícula automática.

31. Crie classe `ContaBancaria` e método `transferir (ContaBancaria destino, double valor)` .

- 32. Crie classe `Produto` com desconto percentual privado.
- 33. Crie método `aplicarDesconto(double percentual)` com validação.
- 34. Crie classe `Lampada` com brilho privado.
- 35. Crie método para aumentar/diminuir brilho dentro de limites.
- 36. Crie classe `Televisao` com canal privado.
- 37. Crie método para mudar canal com validação.
- 38. Crie classe `Relogio` com hora privada.
- 39. Crie método para ajustar hora com validação.
- 40. Crie classe `Pedido` e método para calcular total usando encapsulamento.

Módulo 3 – Herança e Polimorfismo (81 a 140)

1. Crie uma classe `Veiculo` com método `fazerSom()` e herde as classes `Cachorro` e `Gato` sobrescrevendo o método.
2. Crie método `mover()` e herde `Carro` e `Bicicleta` .
3. Crie uma classe `Funcionario` e herde `Gerente` adicionando atributo `bonus` .
4. Crie classe `Funcionario` e herde `Vendedor` adicionando atributo `comissao` .
5. Crie uma classe `FormaGeometrica` com método `calcularArea()` e herde `Quadrado` e `Circulo` .
6. Crie classe `Pessoa` e herde `Aluno` e `Professor` .
7. Crie método `imprimirDados()` nas classes filhas usando polimorfismo.
8. Crie uma classe `ContaBancaria` e herde `ContaCorrente` e `ContaPoupanca` .
9. Crie método `sacar()` sobrescrito na `ContaPoupanca` .
10. Crie método `sacar()` sobrescrito na `ContaCorrente` considerando limite.

11. Crie uma classe `Funcionario` com método `calcularSalario()` .
12. Herde `Gerente` e `Vendedor` sobrescrevendo `calcularSalario()` .
13. Crie classe `Veiculo` com atributo `velocidade` e herde `Carro` e `Moto` .
14. Crie classe `Pessoa` e herde `Cliente` e `Fornecedor` .
15. Crie método `imprimirInfo()` nas classes filhas.
16. Crie classe `Funcionario` e herde `Estagiario` adicionando atributo `horas` .
17. Crie método `calcularSalario()` considerando horas do estagiário.
18. Crie classe `Eletronico` e herde `Celular` e `Notebook` .
19. Crie método `ligar()` sobrescrito nas classes fil

Módulo 4 – Interfaces e Classes Abstratas (141 a 170)

1. Crie interface `Operacoes` com método `calcular()` .
2. Crie classe `Soma` implementando `Operacoes` .
3. Crie classe `Multiplicacao` implementando `Operacoes` .
4. Crie interface `Animal` com método `som()` .
5. Crie classes `Cachorro` e `Gato` implementando a interface.
6. Crie interface `Veiculo` com método `mover()` .
7. Crie classes `Carro` e `Bicicleta` implementando a interface.
8. Crie interface `Pagamento` com método `pagar(double valor)` .
9. Crie classes `Cartao` e `Dinheiro` implementando `Pagamento` .

10. Crie interface `Relatorio` com método `gerarRelatorio()` .

11. Crie classes `RelatorioPDF` e `RelatorioHTML` implementando `Relatorio` .

12. Crie interface `Figura` com método `desenhar()` .

13. Crie classes `Quadrado` e `Circulo` implementando `Figura` .

14. Crie interface `Sensor` com método `ler()` .

15. Crie classes `SensorTemperatura` e `SensorPressao` .

16. Crie interface `Conexao` com método `conectar()` .

17. Crie classes `WiFi` e `Bluetooth` implementando `Conexao` .

18. Crie interface `Armazenamento` com método `salvar()` .

19. Crie classes `HD` e `SSD` implementando `Armazenamento` .

20. Crie interface `Servico` com método `executar()` .

21. Crie classes `Backup` e `Restauracao` implementando `Servico` .

22. Crie interface `PagamentoOnline` com método `processar()` .

23. Crie classes `Paypal` e `Stripe` implementando `PagamentoOnline` .

24. Crie interface `Documento` com método `abrir()` .

25. Crie classes `PDF` e `Word` implementando `Documento` .

26. Crie interface `Transporte` com método `viajar()` .

27. Crie classes `Carro` e `Aviao` implementando `Transporte` .

28. Crie classe abstrata `Funcionario` com método abstrato `()` .

29. Crie classes `Gerente` e `Vendedor` estendendo `Funcionario` . `calcularSalario`

30. Crie classe abstrata `Conta` com método `sacar()` . Crie e `ContaPoupanca` . `ContaCorrente`

Módulo 5 – Exercícios Práticos e Aplicações (171 a 200)

1. Crie sistema de cadastro de `Alunos` com lista e métodos de adicionar/remover.
2. Crie sistema de gerenciamento de `Produtos` com estoque e métodos de adicionar/remover.
3. Crie sistema de `Banco` com contas e transferências.
4. Crie sistema de `Biblioteca` com cadastro de livros e empréstimos.
5. Crie sistema de `Pedidos` com itens e cálculo de total.
6. Crie sistema de `Funcionarios` com aumento salarial e relatórios.
7. Crie sistema de `Veiculos` com aceleração e frenagem.
8. Crie sistema de `Animais` com sons e alimentação.
9. Crie sistema de `Telefones` com ligações recebidas e efetuadas.

10. Crie sistema de `Lampadas` com ligar, apagar e ajustar brilho.

11. Crie sistema de `Clientes` com CPF e validação.

12. Crie sistema de `Estoque` com produtos e quantidade mínima.

13. Crie sistema de `Pedidos` com cálculo de imposto.

14. Crie sistema de `Relatorios` com diferentes formatos (PDF, HTML).

15. Crie sistema de `Pagamento` com cartão, dinheiro e transferência.

16. Crie sistema de `Veiculos` com herança (Carro, Moto, Caminhão).

17. Crie sistema de `ContaBancaria` com polimorfismo para diferentes tipos de conta.

18. Crie sistema de `FormaGeometrica` com cálculo de área polimórfico.
19. Crie sistema de `Animais` com interface `Som` implementada em várias classes.
20. Crie sistema de `Sensores` com interface e classes concretas.
21. Crie sistema de `Conexoes` com interface `Conectar` .
22. Crie sistema de `Documentos` com interface `abrir()` .
23. Crie sistema de `Servicos` com interface `executar()` .
24. Crie sistema de `PagamentoOnline` com diferentes gateways.
25. Crie sistema de `Transporte` com métodos polimórficos.
26. Crie aplicação de cadastro de `Alunos` usando herança (`AlunoRegular`, `AlunoBolsista`).

27. Crie aplicação de `Banco` usando classes abstratas e interfaces.

28. Crie aplicação de `Pedido` com interface `Pagamento` implementada por múltiplos métodos.

29. Crie aplicação de `Veiculos` com interface `Movimentar` implementada por diferentes tipos.

30. Crie sistema completo de `Biblioteca` com classes abstratas, interfaces, polimorfismo e herança.